

Frame Representations for Texture Segmentation

Andrew Laine and Jian Fan

Abstract— We introduce a novel method of feature extraction for texture segmentation that relies on multi-channel wavelet frames and two-dimensional envelope detection. We describe and compare two algorithms for envelope detection based on (1) the Hilbert transform and (2) zero-crossings. We present criteria for filter selection and discuss quantitatively their effect on feature extraction. The performance of our method is demonstrated experimentally on samples of both natural and synthetic textures.

Keywords— **Feature extraction, image segmentation, wavelet analysis.**

I. INTRODUCTION

Features for texture representation are of crucial importance for accomplishing segmentation[1]. Previous multi-channel approaches for texture feature extraction utilized the concept of spatial-frequency representation [2] [3], and have been supported by studies of the human visual system [4]. In these methods, both complex and real filters were used. Complex prolate spheroidal sequences were used as channel filters in [5], and channel envelopes were extracted to form a feature vector. In another study [6], complex Gabor filters were applied, and envelope and phase information were extracted from two quadratic components of distinct output channels. Use of real Gabor filters was reported in [7], where feature extraction included a non-linear transformation, $\psi(t) = \tanh(\alpha t)$, and a statistical measurement of average absolute derivation was performed on overlapping windows. However, even this approach of feature extraction had limitations [7].

Although Gabor filters possess desirable properties for this application, recent developments in wavelet theory provide an alternative approach with several advantages: (1) Wavelet filters cover *exactly* the frequency domain (provide a mathematically complete representation), (2) Correlations between features extracted from distinct filter banks can be greatly reduced by selecting appropriate filters, (3) Adaptive pruning of a decomposition tree makes possible the reduction of computational complexity and the length of feature vectors and, (4) Fast algorithms are readily available to facilitate implementation. In addition, recent studies have reported the success of applying wavelet theory to problems in texture analysis [8][9][10][11][12].

In this correspondence, we adopt real wavelet packet frames (tree structured filter banks [13]) for channel filters, and introduce two envelope detection algorithms for feature extraction. The performance of the two algorithms are then analyzed and compared.

II. MULTI-CHANNEL WAVELET ANALYSIS

An overcomplete wavelet representation called a discrete wavelet packet frame (DWPF) is similar to a discrete

wavelet packet transform [14][15] except that no down sampling occurs between levels of analysis. Figure 1 shows a general DWPF as a binary tree for a three level decomposition. Although a frame representation may be inefficient, we claim that it has several advantages: (1) less restrictions on filter selection, (2) no aliasing, and (3) translation invariance. The last two properties are especially desirable for signal representation and analysis.

For the structure shown in Figure 1, filters $H_l(\omega)$ and $G_l(\omega)$ at level l were generated as described in [16] : $H_l(\omega) = H_0(2^l\omega)$, $G_l(\omega) = G_0(2^l\omega)$. Let $S_k^l(\omega)$ be the Fourier transform of the frame coefficients at channel k for level l , then $S_{2k}^{l+1}(\omega) = G_l(\omega)S_k^l(\omega)$, $S_{2k+1}^{l+1}(\omega) = H_l(\omega)S_k^l(\omega)$.

This is equivalent to a filter bank with channel filters $\{F_k^l(\omega) | 0 \leq k \leq 2^l - 1\}$, where $F_k^l(\omega)$ is defined recursively by the formula $F_0^0(\omega) = G_0(\omega)$, $F_1^0(\omega) = H_0(\omega)$, $F_{2k}^{l+1}(\omega) = G_{l+1}(\omega)F_k^l(\omega) = G_0(2^{l+1}\omega)F_k^l(\omega)$, $F_{2k+1}^{l+1}(\omega) = H_{l+1}(\omega)F_k^l(\omega) = H_0(2^{l+1}\omega)F_k^l(\omega)$. It can be shown that for filters $H_0(\omega)$ and $G_0(\omega)$ satisfying

$$|H_0(\omega)|^2 + |G_0(\omega)|^2 = 1, \text{ or, } |H_0(\omega)| + |G_0(\omega)| = 1, \quad (1)$$

the filter bank $\{F_k^l(\omega)\}$ covers exactly the frequency domain: $\sum_{k=0}^{2^l-1} |F_k^l(\omega)|^2 = 1$, or $\sum_{k=0}^{2^l-1} |F_k^l(\omega)| = 1$.

For images, we simply use a tensor product extension for which the channel filters are written as $F_{i \times j}^l(\omega_x, \omega_y) = F_i^l(\omega_x)F_j^l(\omega_y)$. Such 2-D filters naturally exhibit orientation selectivity. We classified each node in the decomposition tree into four possible categories taking into account orientation:

- The root node is *omni-directional*.
- The node last filtered by $G_l(\omega_x)H_l(\omega_y)$ corresponds to *vertical-orientation*. (High-pass filter G_l is applied row-wise and low-pass filter H_l column-wise.)
- The node last filtered by $H_l(\omega_x)G_l(\omega_y)$ corresponds to *horizontal-orientation*. (Low-pass filter H_l is applied row-wise and high-pass filter G_l column-wise.)
- The node last filtered by $G_l(\omega_x)G_l(\omega_y)$ corresponds to *diagonal-orientation*. (High-pass filter G_l is applied row-wise and high-pass filter G_l column-wise)
- The node last filtered by $H_l(\omega_x)H_l(\omega_y)$ has the same orientation as its parent. (Low-pass filter H_l is applied row-wise and low-pass filter H_l column-wise)

However, outputs of channel filters are band-pass signals and cannot be used directly as texture features. Further smoothing (grouping) is required, and is discussed next.

III. ENVELOPE DETECTION

In this section, we present two envelope detection algorithms, and investigate their performance. For sake of clarity, we first present each algorithm in the one-dimensional case.

1. Envelope detection by Hilbert transform.

The envelope of a narrowband bandpass signal can be computed by a corresponding analytical signal [17]. For a signal $x(t)$, the analytic signal is defined by: $\tilde{x}(t) = x(t) + j\hat{x}(t)$, where $\hat{x}(t)$ is the Hilbert transform of $x(t)$: $\hat{x}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\eta)}{t-\eta} d\eta$. The envelope of the original signal $x(t)$ is then simply the modulus of the analytic signal $\tilde{x}(t)$: $e(t) = |\tilde{x}(t)| = \sqrt{x^2(t) + \hat{x}^2(t)}$. The Fourier transform of the analytical signal $\tilde{x}(t)$ is:

$$\tilde{X}(\omega) = \begin{cases} 2X(\omega) & , \omega \geq 0 \\ 0 & , \text{otherwise.} \end{cases}$$

Although such frequency characteristics cannot be exactly realized in practice, an approximate FIR Hilbert transformer may be designed by windowing the ideal [18]. In our study, we used a type III FIR Hilbert transformer designed in [18] with parameters $M = 18$, and $\beta = 2.629$. This Hilbert transformer is anti-symmetric, of length 19 and has 10 nonzero coefficients at odd indices.

2. Envelope detection by zero crossings.

In this method, the maximum absolute value between two adjacent zero-crossings was first found, and then assigned to each point within an interval. The algorithm is presented in pseudocode below:

Envelope_ZC($x[1 : N]$)

begin

$i := 1$;

while ($i < N$)

begin

$max = |x[i]|$; $k := i + 1$;

Advance index k while keeping the *maximum absolute value* encountered until next zero-crossing or zero-valued point is found;

Assign value max to all indexes from i to k ;

Update $i := k$;

end

end Envelope_ZC;

Next, we compare the performance of each algorithm. Figure 2 shows examples for input sinusoids spanning a wide range of frequency ($0.01\pi - 0.45\pi$). This clearly demonstrates that the Hilbert transformer performed poorly in very low frequency channels, and was more sensitive to noise perturbations. The noise sensitivity is due to its high-pass nature. In comparison, envelopes detected by zero crossings were less sensitive to noise, and computationally more efficient.

Next we extended the 1-D envelope detection algorithms for the analysis of two-dimensional image signals. In the frequency domain, a two-dimensional analytic signal can be obtained by setting an appropriate half plane to zero, based on its orientation. That is, for a 2-D signal $f(x, y)$, the Fourier transform of an analytic signal $\tilde{f}(x, y)$ is either

$$\tilde{F}(\omega_x, \omega_y) = \begin{cases} 2F(\omega_x, \omega_y) & , \omega_x \geq 0 \\ 0 & , \text{otherwise} \end{cases} \text{ or, } \tilde{F}(\omega_x, \omega_y) = \begin{cases} 2F_{\cos x^1}(\omega_y) & , \frac{|H_0(\omega_1)H_0(\omega_2)| + |G_0(\omega_1)G_0(\omega_2)|}{\sqrt{|H_0(\omega_1)|^2 + |G_0(\omega_1)|^2} \sqrt{|H_0(\omega_2)|^2 + |G_0(\omega_2)|^2}} \\ 0 & , \text{otherwise} \end{cases}$$

For the 2-D filters used in our study, the equivalent complex quadrature filters exhibited the frequency response shown in Figure 3 (Diagonal components can have an alternate arrangement by zeroing out the left half plane).

This separable property allowed us to compute the envelope of a 2-D signal using the 1-D algorithms described earlier in a straightforward manner, described below:

Envelope_2D(W , **Orientation**)

{ W : a 2-D array }

begin

case (**Orientation**)

Horizontal: apply **Envelope_1D** column-wise;

Vertical: apply **Envelope_1D** row-wise;

Diagonal: apply **Envelope_1D** column-wise;

end case;

end Envelope_2D;

Note that no operation was applied to the omnidirectional component due to its low frequency content.

At the end of the feature extraction process, we constructed feature vectors for each pixel: $\vec{V}_{i,j} = \{e_{k,i,j}^L \mid 0 \leq k \leq (2^L - 1)\}$, where $e_{k,i,j}^L$ denotes the envelope value of pixel (i, j) for the k th component at level L in a DWPF tree. This same 2-D envelope detection method is also applicable to filter banks using real Gabor filters [7].

IV. FILTER SELECTION AND FEATURE EXTRACTION

Symmetry, frequency response, and boundary accuracy are important factors in the selection of filters for feature extraction. Below we discuss these constraints in terms of overall performance.

- **Symmetry.** For this application, filters with symmetry or antisymmetry are clearly favored. Such filters have a linear phase response, where the delay (shift) is predictable. Alternatively, filters with nonlinear phase may introduce complex distortion. Moreover, symmetric or anti-symmetric filters are also advantageous in alleviating boundary effects through simple methods of mirror extension.
- **Optimal frequency response.** In order to derive an ideal filter frequency response, we considered a two-band filter bank with input signals of infinite length consisting of two segments with distinct pure tones. The input signals can be written as

$$s(n) = \begin{cases} A_1 \cos(\omega_1 n + \alpha_1) & , n < 0, \\ A_2 \cos(\omega_2 n + \alpha_2) & , n \geq 0, \end{cases} \quad (2)$$

where $A_1 > 0$ and $A_2 > 0$. Except for the boundary ($n = 0$), we derived the feature vectors (envelopes of channel outputs) as,

$$\vec{T} = (e_H, e_G) = \begin{cases} \vec{T}_{left} = (A_1 |H_0(\omega_1)|, A_1 |G_0(\omega_1)|) & , n < 0, \\ \vec{T}_{right} = (A_2 |H_0(\omega_2)|, A_2 |G_0(\omega_2)|) & , n \geq 0. \end{cases}$$

The angle θ between vectors \vec{T}_{left} and \vec{T}_{right} is

and is bound by $0 \leq \theta \leq \frac{\pi}{2}$. The distance between the two classes in the feature space is then

$$D = \sqrt{\left| \vec{T}_{left} \right|^2 + \left| \vec{T}_{right} \right|^2 - 2 \left| \vec{T}_{left} \right| \left| \vec{T}_{right} \right| \cos(\theta)} .$$

For $\theta = \frac{\pi}{2}$, vectors \vec{T}_{left} and \vec{T}_{right} are orthogonal, and the distance D reaches its maximum. Clearly, the maximum distance in feature space between any two classes is optimal for segmentation and classification applications, in the sense that the classes are better separated and more robust to noise perturbations. Notice that for $H_0(\omega_1) \neq 0$ and $G_0(\omega_2) \neq 0$, $\cos(\theta) = 0$ if and only if $G_0(\omega_1) = 0$ and $H_0(\omega_2) = 0$. This means that optimal filter banks should have no overlap in the frequency domain, and thus filter $H_0(\omega)$ with optimal frequency response must be a perfect half-band filter. Indeed, [19] derived a similar optimal filter for image coding applications. Thus, we can connect orthogonality of the feature set to the property of a filter bank. For any pair of signal segments as defined in Equation (2) with pure tones (ω_1, ω_2) and $H_0(\omega_1) \neq 0$ and $G_0(\omega_2) \neq 0$, if the feature vector set corresponding to the two signal segments are orthogonal, we define the two filter bands to be orthogonal. Of course, this concept can be extended to multi-band filter banks. For a tree-structured filter bank shown in Figure 1, any two bands at the same level are orthogonal if and only if the two bands at level 0 are orthogonal. However, such ideal filters cannot be realized in practice. Based on the discussion above, filter bands overlapping in the frequency domain tend to *reduce* class distance in feature space. Therefore, a filter $H_0(\omega)$ with a large stop-band attenuation and flat pass-band frequency response is desirable.

- **Spatial localization.** There are two types of boundaries for signals of finite length: (1) boundaries of regions exhibiting distinct characteristics and (2) the physical boundaries of a data segment. Feature vectors close to the boundaries will be effected. The size of the effected region depends on the length of the channel filters, and the distribution of filter coefficients. Therefore, filters of short length and fast decay shall be better suited for boundary detection.

Unfortunately, the above three criteria cannot be satisfied simultaneously. Quadrature Mirror Filters (QMF) satisfy the exact coverage constraint (Equation (1)), but, real QMF's with compact support cannot be symmetric or anti-symmetric [20]. Moreover, large attenuation in the stop band requires a longer filter, which in turn degrades the filter's spatial localization. Therefore, the best filter selection is a trade-off among these criteria and the characteristics of an input signal. That is, if classes in a signal differ mainly in frequency characteristics, a filter bank exhibiting good frequency separation should be used. However, if classes in a signal differ mainly in spatial characteristics, a filter bank with good spatial localization should be employed.

An *uncertainty factor* was previously used as an optimal criterion for filter selection [6]. For discrete filters, the uncertainty factor U can be defined as follows [21]:

$$U = \sigma_n^2 \sigma_\omega^2, \quad (3)$$

where $\sigma_\omega^2 = \frac{1}{2\pi E} \int_{-\pi}^{\pi} (\omega - \bar{\omega})^2 |H(e^{j\omega})|^2 d\omega$, $E = \sum_n |h(n)|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(e^{j\omega})|^2 d\omega$, $\bar{n} = \frac{1}{E} \sum_n n |h(n)|^2$, $\bar{\omega} = \frac{1}{2\pi E} \int_{-\pi}^{\pi} \omega |H(e^{j\omega})|^2 d\omega$. For the discrete case, Liu *et al* [21] proved that $U \geq 0.25$. This lower bound is achieved only by Gaussian functions.

However, we were cautious in applying the uncertainty factor as a criterion to evaluate filters for the following reasons: (1) The uncertainty factor is a *product*. Even if two filters have the same factor, their behavior in the spatial and frequency domains may be quite distinct. (2) Channel filters in a filter bank may have different values of U . (3) Prototype filter $H_0(\omega)$ must respect the constraint of Equation (1). As discussed above, the ideal filter for $H_0(\omega)$ is a perfect half-band low-pass filter. Therefore, we used the maximum value of U_{max} in each filter bank for comparison. In addition, rather than evaluating a filter bank at the top level alone, we evaluated each filter bank at several levels close to the bottom of each tree.

Finally, we compared filters which satisfied the linear phase constraint and Equation (1):

(1) **Lemarié-Battle filters** [22]. This is a class of symmetric quadrature mirror filters (QMF) with infinite length. The high-pass filter $g_0(n)$ was obtained by $g_0(n) = (-1)^n h_0(n)$, or, $G_0(\omega) = H_0(\omega + \pi)$. Thus, both low-pass and high-pass filters were symmetric. In practice, we truncate each to finite length.

(2) **Autocorrelation shell** [23]. For a quadrature mirror filter $Q(\omega)$ satisfying $|Q(\omega)|^2 + |Q(\pi + \omega)|^2 = 1$, we may define discrete filters $H_0(\omega)$ and $G_0(\omega)$ from $Q(\omega)$ as $H_0(\omega) = |Q(\omega)|^2$ and $G_0(\omega) = |Q(\pi + \omega)|^2$. Filters $H_0(\omega)$ and $G_0(\omega)$ are both symmetric. For FIR $Q(\omega)$, $H_0(\omega)$ and $G_0(\omega)$ are both FIRs. In our study, both Lemarié-Battle filters and FIR Daubechies filters were applied.

The results of the comparison are presented in Figure 4. Figures 4 (a) and (b) show the maximum uncertainty factor U_{max} at levels 4, 5 and 6 *vs.* filter order for Lemarié-Battle filters and an autocorrelation shell of Lemarié-Battle filters, respectively. 1-D filter banks were evaluated using a 512-point FFT. We observed that filter orders corresponding to minimum U_{max} were not the same at distinct levels. Also deep in the tree, the higher order filters (longer length) corresponded to minimum U_{max} . Figure 4 (c) shows U_{max} at levels 4, 5 and 6 *vs.* filter length for the autocorrelation shell of Daubechies wavelet filters. Note that for longer filters in the range of 4 – 20 taps, we observed a consistent decreasing of U_{max} values.

V. EXPERIMENTAL RESULTS FOR MULTI-CHANNEL TEXTURE SEGMENTATION

We tested our representation using an *ISODATA* clustering algorithm [24]. The number of distinct classes in each

textured image was a required parameter for the program. Our test images included samples of two distinct families of texture:

- **Natural textures.** Here we used textures obtained from the Brodatz album [25] and public archive. Each testing sample was first histogram equalized so that a segmentation result based only on first order statistics was not possible. Experimental results are displayed in Figure 6. Experimentally we observed that a lower order Lemarié-Battle filter (order 1) performed well in boundary detection (Figure 6 (b)), while the higher order Lemarié-Battle filter (order 5) did a better job within non-boundary (internal) regions.
- **Synthetic textures.** We also tested the performance of our algorithm on several synthetic (computer generated) images of texture. Figure 7 shows a segmentation result on a Gaussian low-pass texture image [1], and Figure 8 shows a segmentation result on a filtered impulse noise (FIN) texture image [1]. For this difficult test image, the algorithm achieved “super-human” performance. Figure 9 demonstrates an accurate segmentation result for a texture image containing regular and sparse elements.

A quantitative comparison, presenting the accuracy of our segmentation results is summarized in Table II. This performance is consistent with the difficulty of segmentation perceived by human observers. We observed that boundary errors were dependent on shape *i.e.*, complex boundaries yielded more variance.

VI. CONCLUSIONS

We have described a feature extraction method for texture segmentation that relied upon (overcomplete) wavelet packet frame representations. Two approaches for accomplishing one-dimensional envelope detection were presented and formulated for two-dimensional analysis. In addition, performance measures for reliable texture segmentation were evaluated. We then discussed criteria for filter selection and presented some quantitative comparisons. Finally, we presented convincing experimental results for several distinct texture types including macro-texture and micro-textures.

Future research shall focus on exploiting the multiresolution nature of this representation to increase the accuracy of class boundaries, and finding an adaptive and robust criterion to prune the decomposition tree.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under Grant No. IRI-9111375. The authors thank Iztok Koren for his valuable discussions and comments during revision of the manuscript.

REFERENCES

- [1] M.Kardan J.M.H.DU Buf and M.Span, “Texture feature performance for image segmentation”, *Pattern Recognition*, vol. 23, pp. 291–309, 1990.
- [2] J.Y.Hsiao and A.A.Sawchuk, “Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques”, *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 11, pp. 1279–1292, 1989.
- [3] T.R.Reed and H. Wechesler, “Segmentation of textured images and gestalt organization using spatial/spatial-frequency representations”, *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 12, pp. 1–12, 1990.
- [4] S. G. Mallat, “Multifrequency channel decompositions of images and wavelet methods”, *IEEE Transaction on Acoustics, Speech and Signal Processing*, vol. 37, pp. 2091–2110, 1989.
- [5] R. Wilson and M. Spann, “Finite prolate spheroidal sequences and their applications ii: image feature description and segmentation”, *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 10, pp. 193–203, 1988.
- [6] A. C. Bovik, “Analysis of multichannel narrowband filters for image texture segmentation”, *IEEE Transaction on Signal Processing*, vol. 39, pp. 2025–2043, Sept. 1991.
- [7] A. K. Jain and F. Farrokhnia, “Unsupervised texture segmentation using gabor filters”, *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.
- [8] A. Laine and J. Fan, “An adaptive approach for texture segmentation by multichannel wavelet frames”, in *SPIE Proceedings on Mathematical Imaging: Wavelet Applications in Signal and Image Processing*, 1993, vol. 2034, pp. 288–299.
- [9] A. Laine and J. Fan, “Texture classification by wavelet packet signatures”, *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 15, no. 11, pp. 1186–1191, 1993.
- [10] A. Kundu and J. Chen, “Texture classification using qmf bank-based subband decomposition”, *CVGIP: Graphical Models and Image processing*, vol. 54, no. 5, pp. 369–384, 1992.
- [11] J. Chen and A. Kundu, “Rotation and gray scale transform invariant texture identification using wavelet decomposition and hidden markov model”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 208–214, 1993.
- [12] M. Unser, “Texture discrimination using wavelets”, in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1993, pp. 640–641.
- [13] P. P. Vaidyanathan, *Multirate systems and filter banks*, Prentice-Hall, 1993.
- [14] M. V. Wickerhauser, “Acoustic signal compression with wavelet packets”, in *Wavelets: a tutorial in theory and applications*, C. K. Chui, Ed. Academic Press, 1992.
- [15] R. R. Coifman and M. V. Wickerhauser, “Entropy-based algorithms for best basis selection”, *IEEE Transactions on Information Theory*, vol. 38, pp. 713–718, 1992.
- [16] O. Rioul and M. Vetterli, “Wavelet and signal processing”, *IEEE Signal Processing Magazine*, pp. 14–38, Oct. 1991.
- [17] A. Papoulis, *Signal analysis*, McGraw-Hill Book Company, 1977.
- [18] A. V. Oppenheim and R. W. Schaffer, *Discrete-time signal processing*, Prentice Hall, 1989.
- [19] B. Macq P. Desarte and D. T. Slock, “Signal-adapted multiresolution transform for image coding”, *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 897–904, 1992.
- [20] I. Daubechies, *Ten lectures on wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1992.
- [21] Yipeng Liu and A.N. Akansu, “An evaluation of time-frequency localization in transforms and filter banks”, in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 1993, pp. 261–263.
- [22] S. Mallat, “A theory for multiresolution signal decomposition : the wavelet representation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, 1989.
- [23] Naoki Saito, *Local feature extraction and its applications using a library of bases*, PhD thesis, Yale University, 1994.
- [24] R. O. Duda and P. E. Hart, *Pattern classification and scene analysis*, John Wiley & Sons, 1973.
- [25] P. Brodatz, *Textures-a photographic album for artists and designers*, Dover, 1966.

TABLE I

BOUNDARY ACCURACY FOR MULTI-CHANNEL SEGMENTATION.

Test image	Maximum ABE	Average ABE	$\pm\sigma$
T1	10.0	2.9	2.6
T2	14.0	2.6	2.1
T3	12.0	2.7	2.3

ABE: Absolute Boundary Error (in pixels).

TABLE II
BOUNDARY ACCURACY FOR MULTI-CHANNEL SEGMENTATION.

Test image	Maximum ABE	Average ABE	$\pm\sigma$
T1	10.0	2.9	2.6
T2	14.0	2.6	2.1
T3	12.0	2.7	2.3

ABE: Absolute Boundary Error (in pixels).

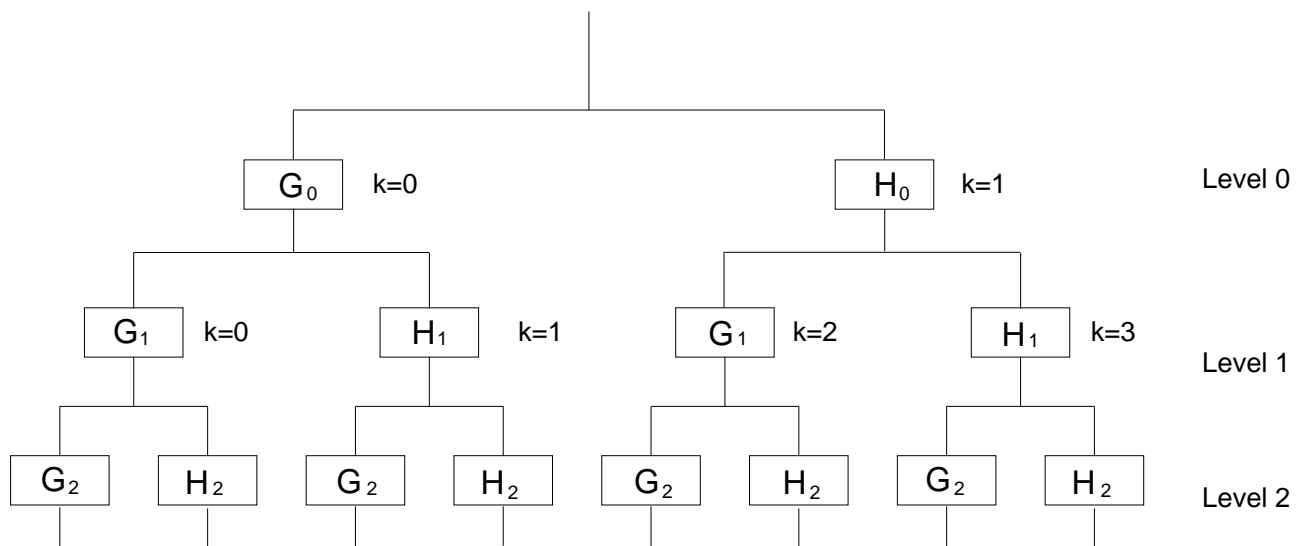


Fig. 1. Tree structure for wavelet packet frames and associated indexes.

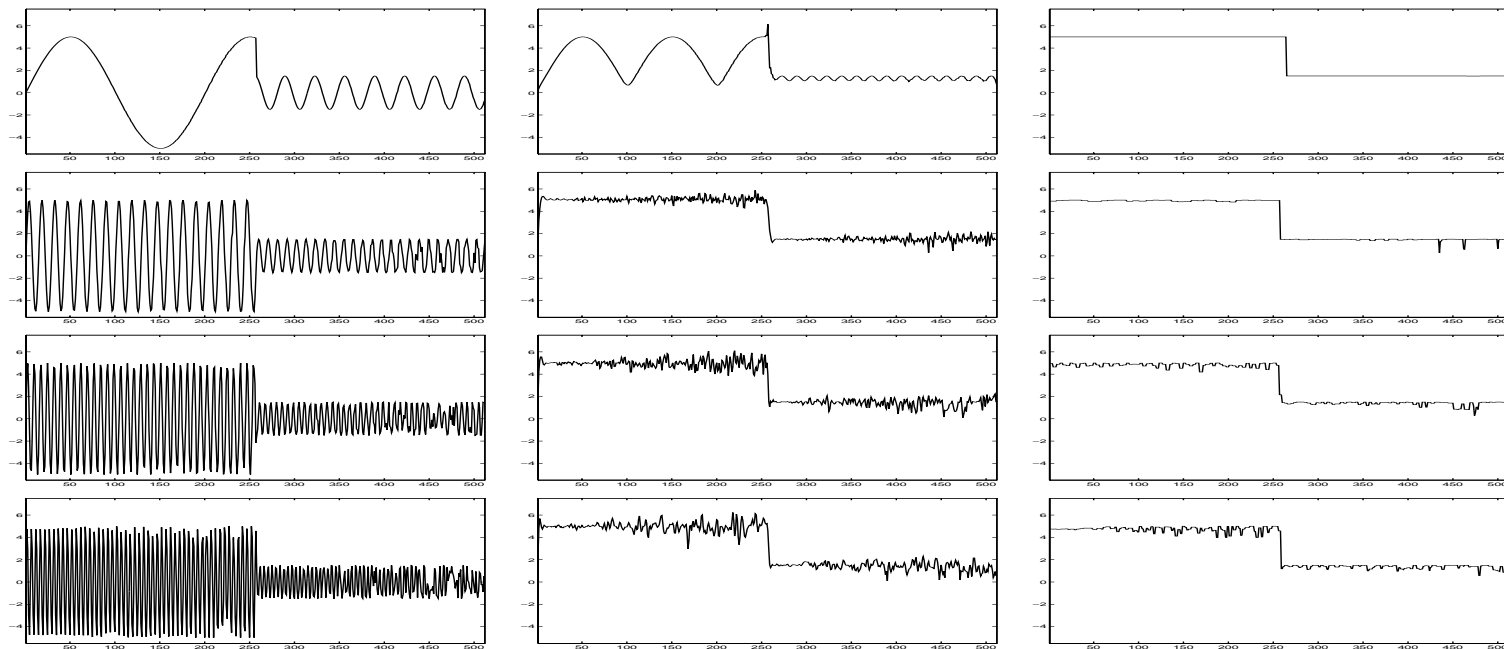


Fig. 2. Comparison of envelope detectors. Column 1: Input signals (sinusoids with frequency perturbation). Column 2: Envelopes detected via a 19-tap Hilbert transformer. Column 3: Envelopes detected via a zero-crossing based method.

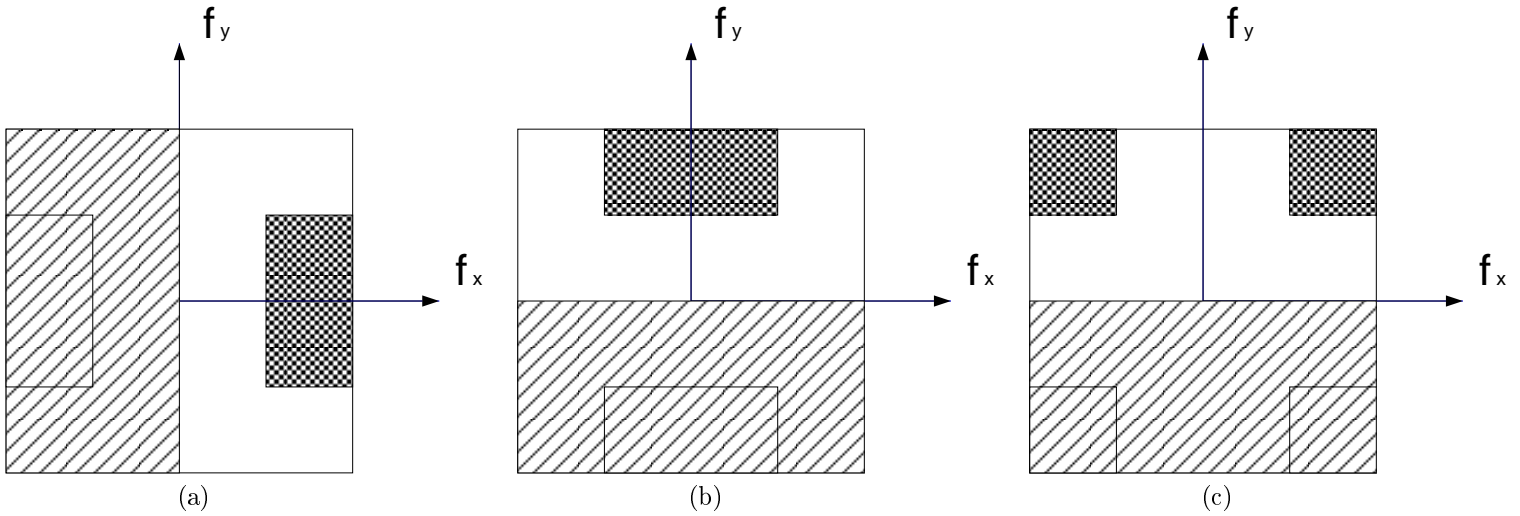


Fig. 3. Frequency response of equivalent complex quadrature filters (level 1) for (a) vertical, (b) horizontal, (c) diagonal components, respectively. The diagonal shadowed areas identify zeroed half planes.

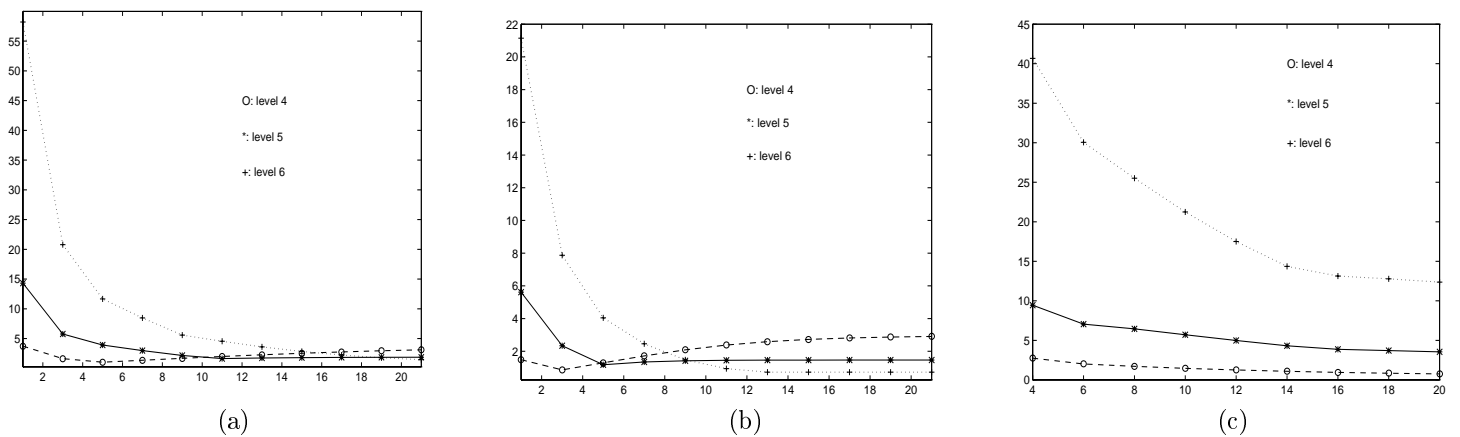


Fig. 4. Comparisons using maximum *uncertainty factor* U_{max} at levels 4, 5 and 6. (a) U_{max} (vertical axis) *vs.* filter order (horizontal axis) for Lemarié-Battle filters. (b) U_{max} (vertical axis) *vs.* filter order (horizontal axis) for autocorrelation shell of Lemarié-Battle filters. (c) U_{max} (vertical axis) *vs.* filter length (horizontal axis) for autocorrelation shell of Daubechies filters.

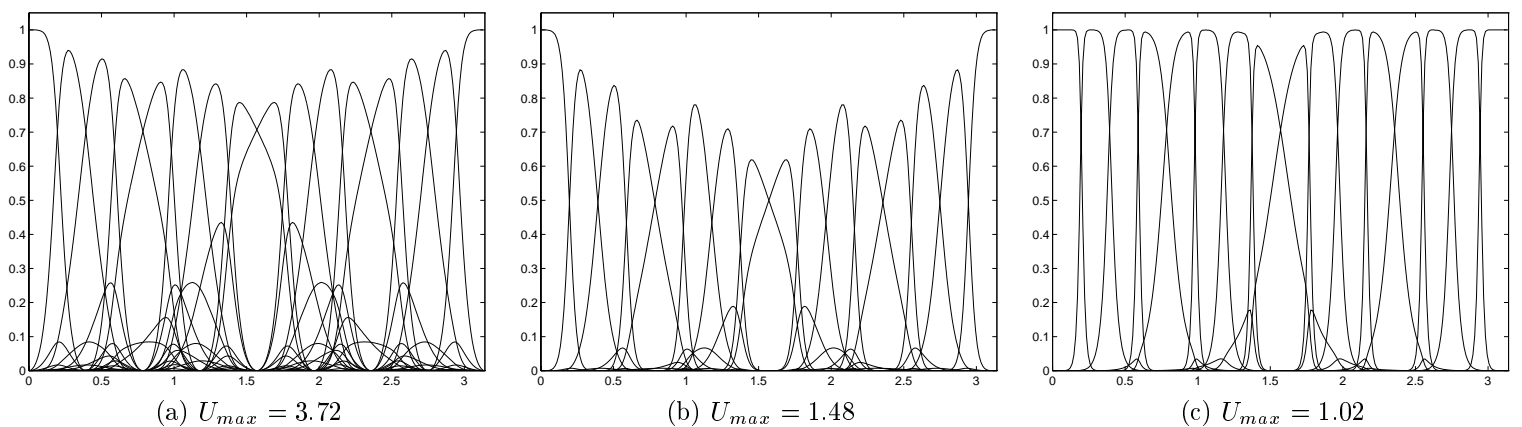


Fig. 5. Filter banks at level 4 generated by (a) Lemarié-Battle filter of order 1. (b) autocorrelation shell of Lemarié-Battle filter of order 1. (c) Lemarié-Battle filter of order 5.

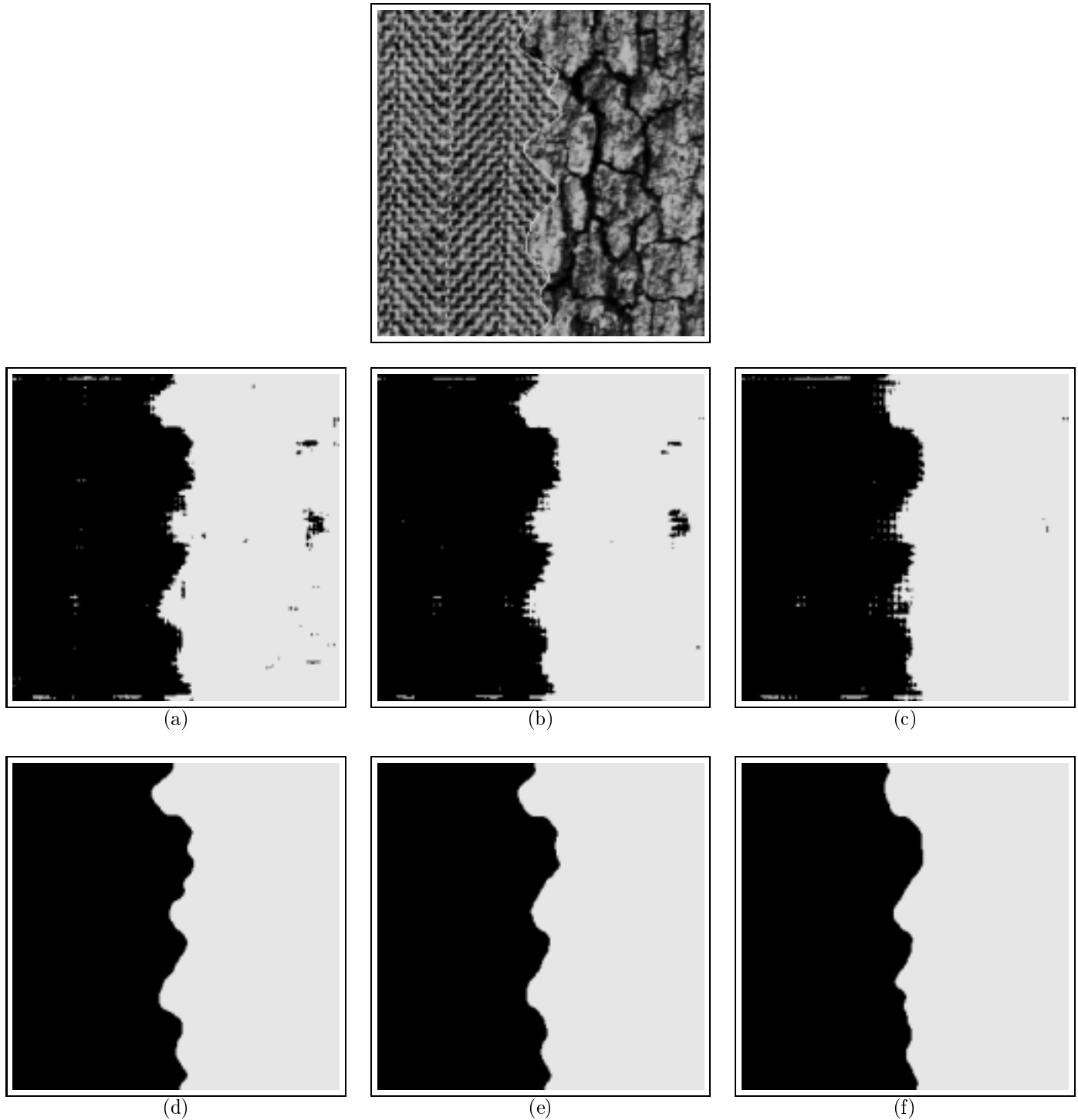


Fig. 6. Comparison of feature extraction using different filters. Top row: Test image T1 (256×256 , 8-bit) consists of D17, herringbone weave and bark (true boundary overlaid for display only). Middle row: Clustering results using features extracted from a level 4 filter bank generated by (a) Lemarié-Battle filter of order 1, filter bank $U_{max} = 3.72$. (b) autocorrelation shell of Lemarié-Battle filter of order 1, filter bank $U_{max} = 1.48$. (c) Lemarié-Battle filter of order 5, filter bank $U_{max} = 1.02$. Corresponding 1-D filter banks are shown on Figure 5. The zero-crossing algorithm for envelope detection were used for (a), (b) and (c). Bottom row: Final segmentations after postprocessing.

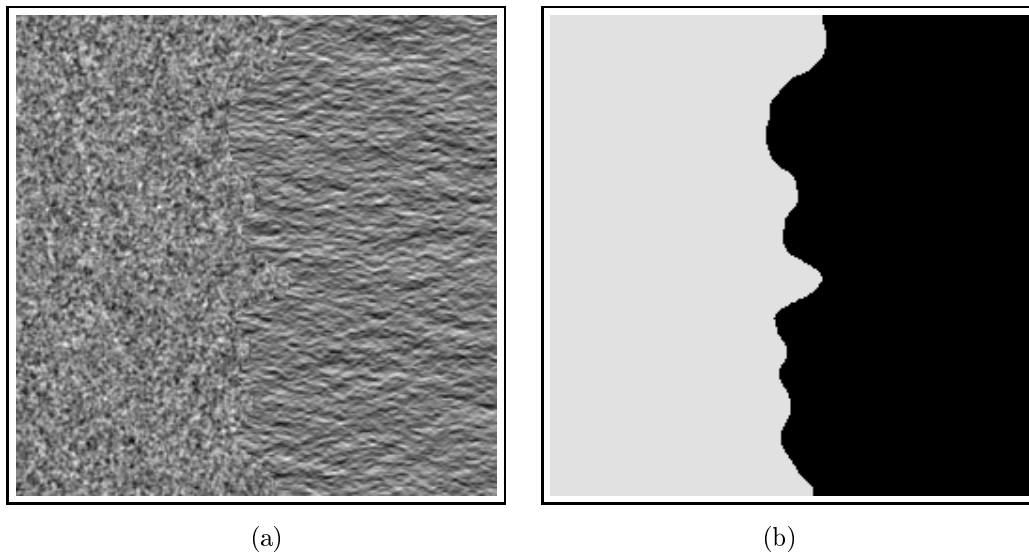


Fig. 7. Segmentation result No.2: (a) Test image T2 (256×256 , 8-bit): Gaussian LP, left: isotropic $F_c = 0, S_r = 60$; right: non-isotropic $F_c = 0, S_r = 60, \theta_0 = 0, B_0 = 0.175$. (b) Final segmentation (Lemarié-Battle filter of order 1, level 4 and zero-crossing envelope detection).

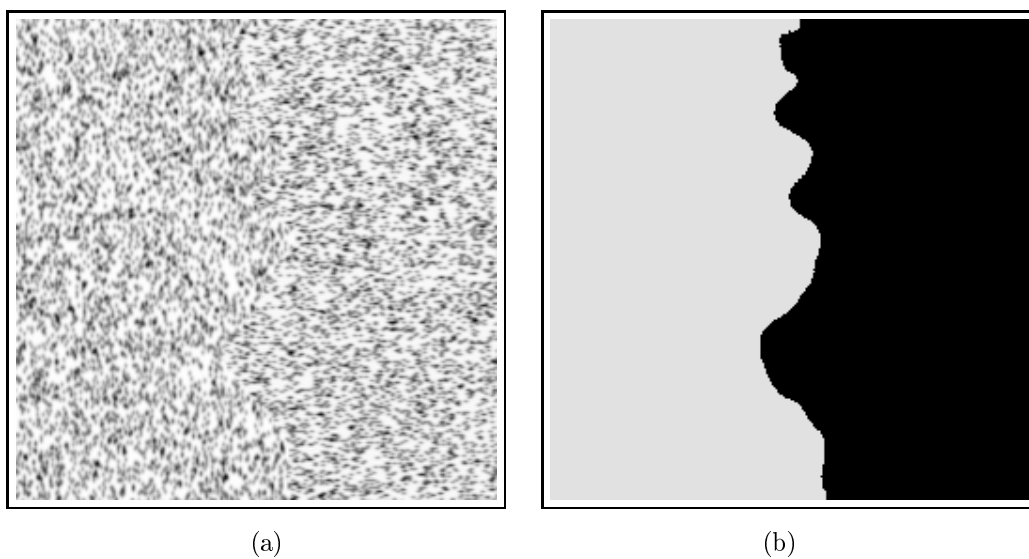


Fig. 8. Segmentation result No.3: (a) Test image T3 (256×256 , 8-bit): Filtered impulse noise, left: non-isotropic $T = 0.15, S_x = 1.0, S_y = 1.5$; right: non-isotropic $T = 0.15, S_x = 2.0, S_y = 1.0$. (b) Final segmentation (Lemarié-Battle filter of order 1, level 4 and zero-crossing envelope detection).

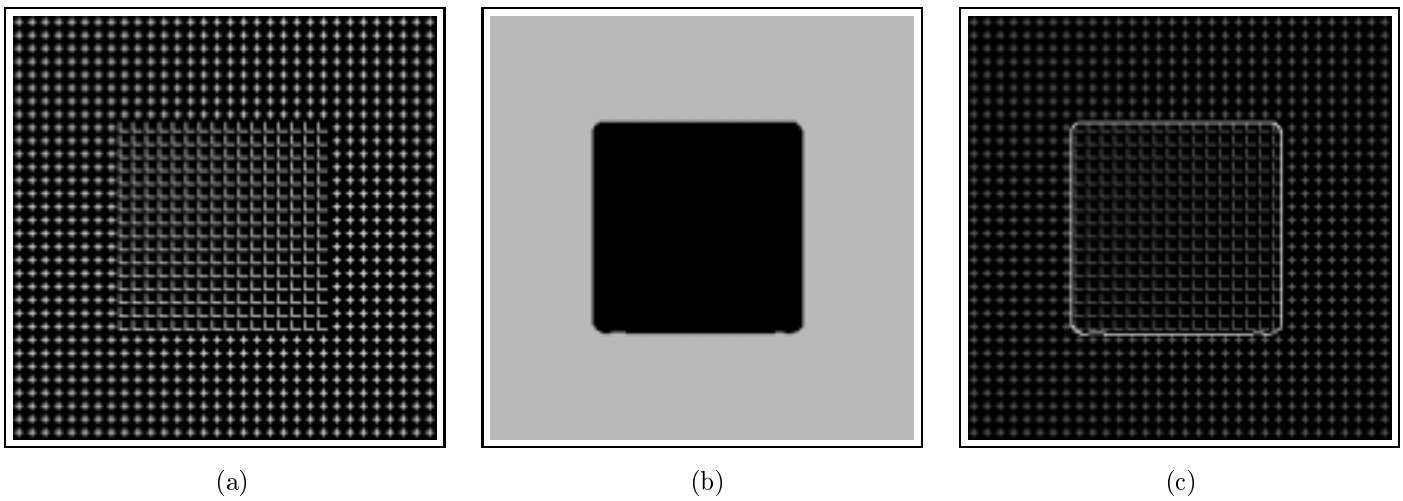


Fig. 9. Segmentation result No.4: (a) Test image T4 (256×256 , 8-bit). (b) Final segmentation (Lemarié-Battle filter of order 1, level 3 and zero-crossing based envelope detection). (c) Detected boundary overlaid with the original image.